

Networking on IBM z17, EQDIO, and z/VM: What it Means for You

(Now With New Features!)



Montana Lee, montanalee@ibm.com
z/VM Networking Developer (EQDIO and VSwitch)
Poughkeepsie, NY

Disclaimer

© Copyright IBM Corporation 2025

IBM and the IBM logo are trademarks of IBM Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on [ibm.com/trademark](https://www.ibm.com/trademark).

This document is current as of the initial date of publication and may be changed by IBM at any time. Statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

Examples presented are illustrative only. Actual results will vary based on client configurations and conditions and, therefore, generally expected results cannot be provided.

Not all offerings are available in every country in which IBM operates.

It is the user's responsibility to verify the operation of any non-IBM products or programs with IBM products and programs. IBM is not responsible for non-IBM products and programs.

The client is responsible for ensuring compliance with all applicable laws and regulations. IBM does not provide legal advice nor represent or warrant that its services or products will ensure that the client is compliant with any law or regulation.

Agenda

4 z17 Networking Overview

- 7 Ethernet Before z17
- 8 Converged Adapter

9 EQDIO and VSWITCH

- 10 Guest Support for Real EQDIO Devices
- 11 VSWITCH Support
- 12 EQDIO or QDIO Dev?
- 13 Migration Notes
- 15 Link Aggregation
- 17 Hipersockets Bridge

18 Feature Pack Additions

- 19 EQDIO MVLAG
- 20 VNIC Simulated EQDIO
- 23 EQDIO Critical Fixes

24 Externals

- 25 Monitor Updates
- 26 Query OSAINFO
- 27 LOG Output

32 TCP/IP on z/VM

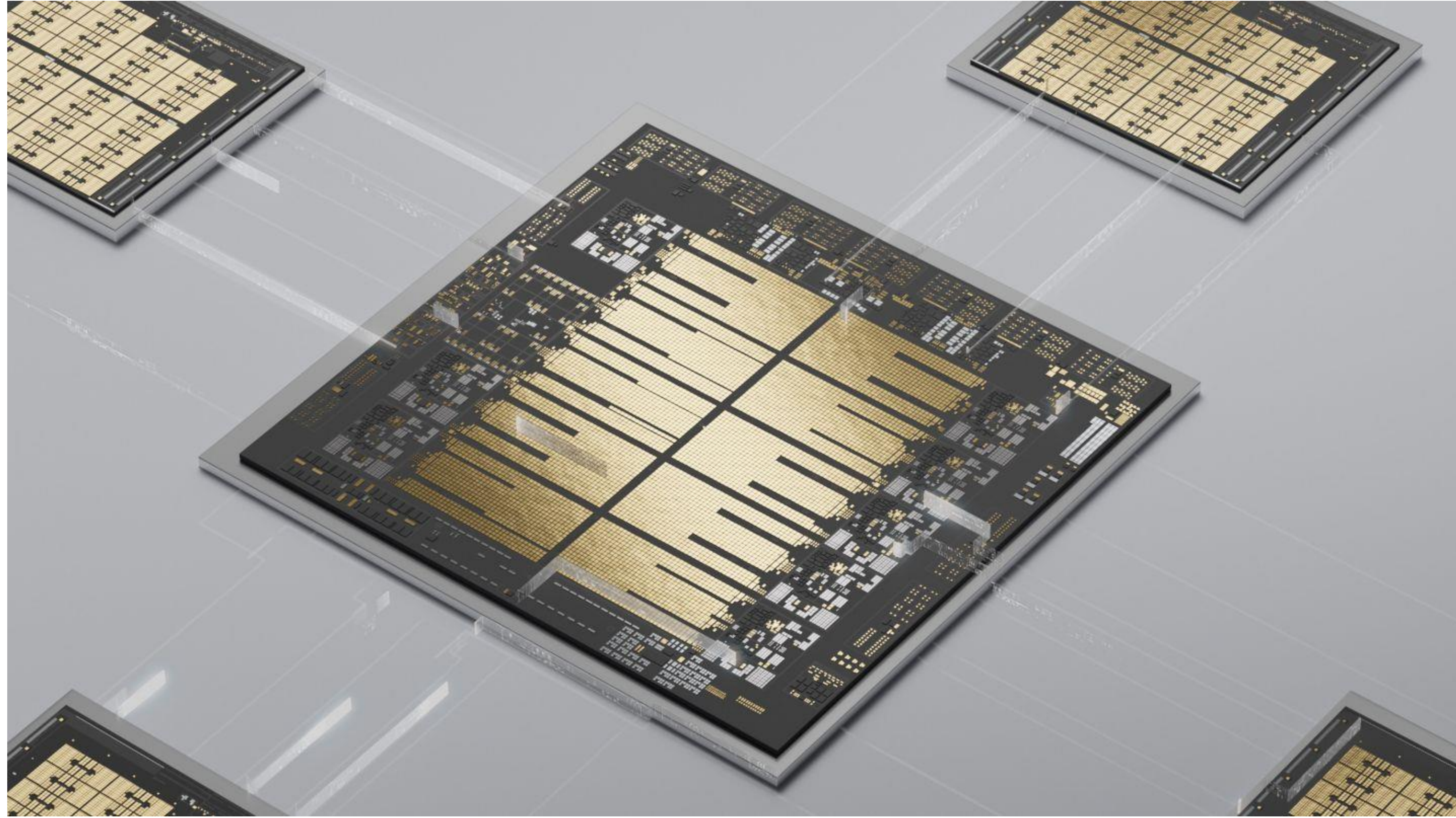
- 33 Configuration
- 36 Migration

38 Summary

40 References

Then, time for *Questions!*
(Or stop me whenever!)

z17 Networking



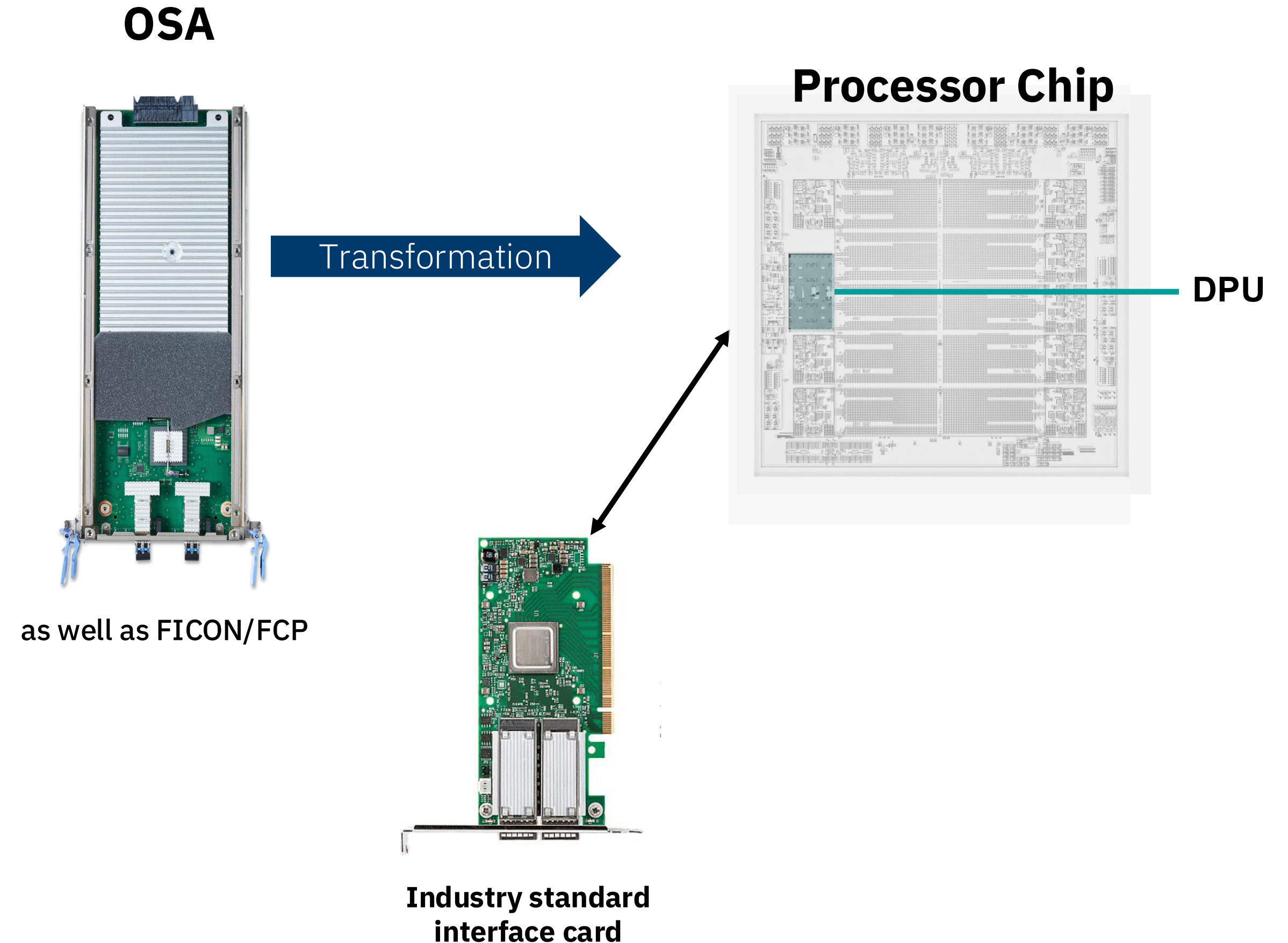
Networking Transformation through I/O Infrastructure Modernization

The **I/O subsystem** of IBM Z is built on a solid architectural basis that **has provided a long and rich history of I/O** (storage and networking) function, throughput and bandwidth improvements, while maintaining application backward-compatibility

The **pre-z17 I/O infrastructure is constrained** in terms of being able to deliver:

- Improved latency,
- Additional bandwidth capability, and
- Enhanced functionality

Without changes to the current physical packaging and (in the case of networking) architecture



Therefore **moving** I/O functionality from the Adapter Card to the processor chip

Hardware Support for I/O Adapter Standardization, Cloud Enablement, & I/O Scalability (Speeds & Feeds)

Z I/O - DPU: Move the I/O abstraction layer (firmware) located within an adapter-based RISC Processor to a zCore, the Data Processing Unit (DPU).

- Locates the adapter closer to Z memory and the nest of Z Cores.
- Supports Legacy Mode FICON, High Performance FICON (HPF), FCP (SCSI over fiber channel), and OSH
- Plus physical function support for Native Ethernet exploitation, allowing for a port to be shared between OSH and PCIe Native protocol

Enhanced QDIO: The next generation abstraction layer for networking.

- OSA-Express adapter MAX out at 14-19 Gbit on a 25 Gbit NIC.
- *EQDIO can reach 25 Gbit*
- Reduce protocol chitchat.
- Cache collision reduction.
- Increased memory efficiency.
- Scalability for Z Virtualization

Ethernet on Z Before z17

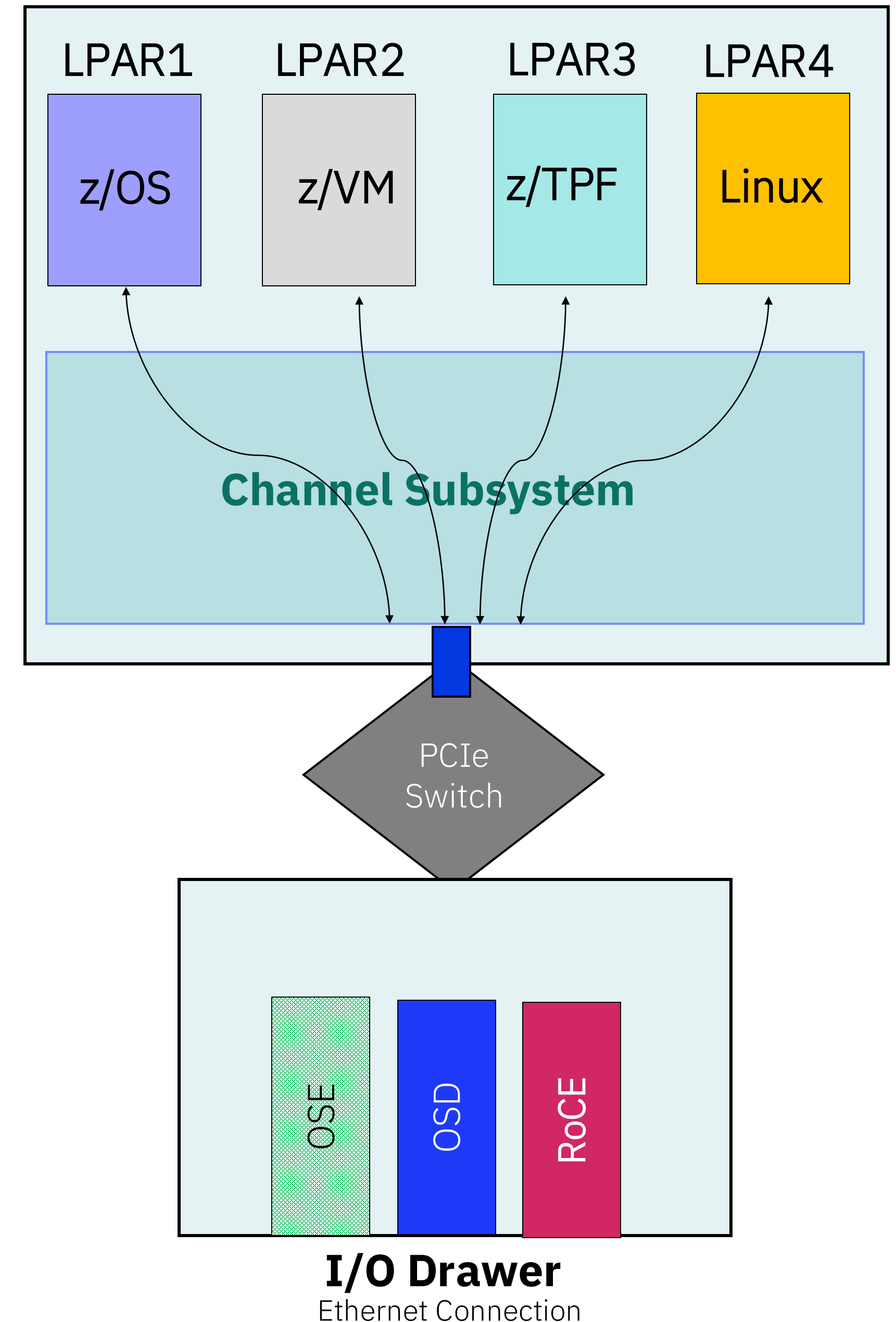
Z has multiple unique functions that ultimately result in some protocol flowing over an Ethernet link

Each protocol requires a unique channel type and a unique networking adapter:

- **OSD** – general TCP/IP, UDP, etc.
- **RoCE** – RDMA capability, used in SMC-R
- **OSE** – SNA (LSA) and TCP/IP (LCS)

When you consider redundancy requirements and your required bandwidth for each protocol for all LPARs, **the pre-z17 system networking design can require many:**

- I/O card slots
- switch cables
- switch ports



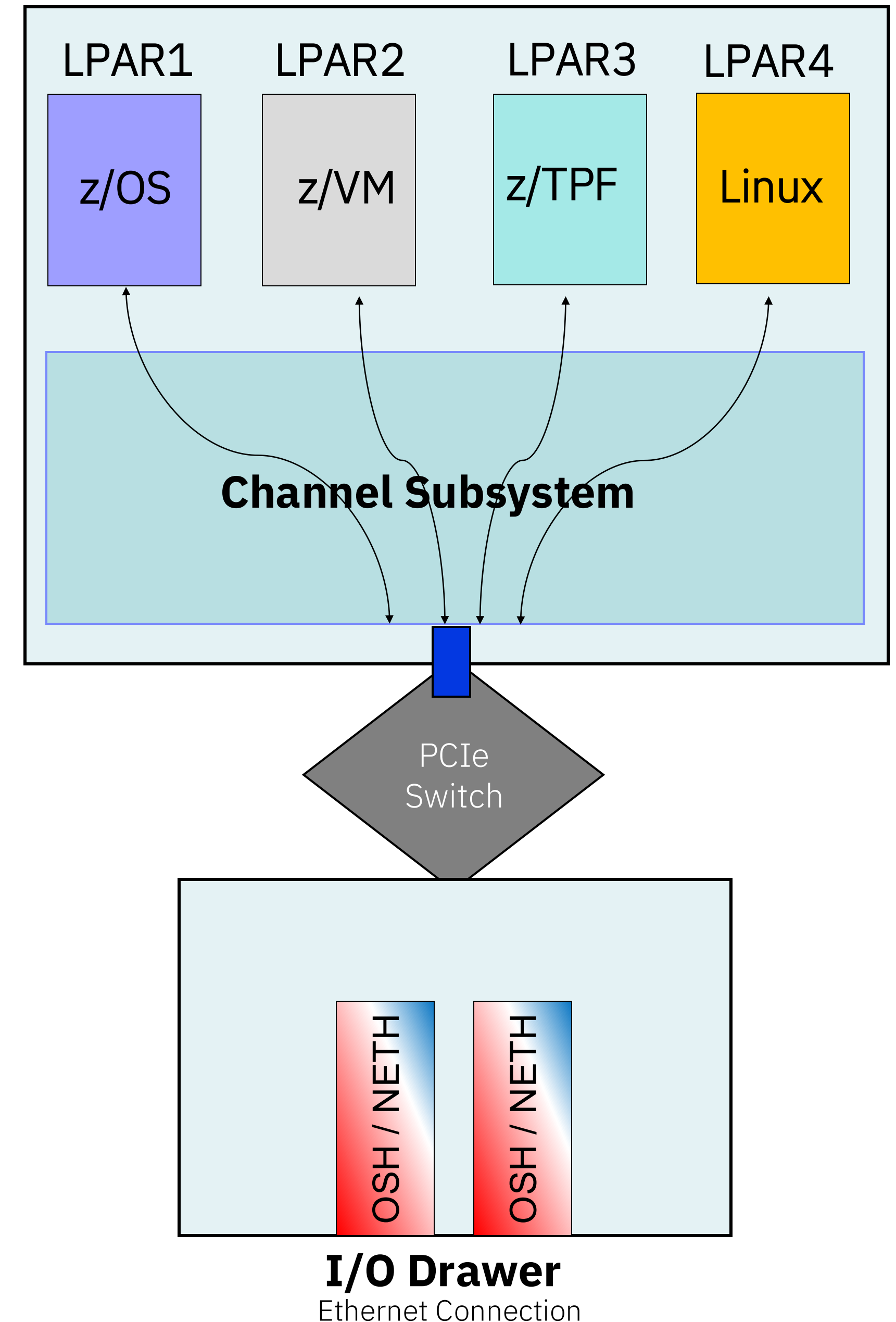
Converged Multi-Function Network Adapter

Provides Converged support for multiple networking protocols – provide ability to run existing functions on single physical “appliance”

- **OSH** – New Enhanced QDIO protocol (EQDIO) capability
- **NETH/RoCE** – RDMA, SMC-R & TCP/IP capabilities

Characteristics

- **Multi-function networking cards have 2 adapters**
 - Initial support for 10GbE, 25GbE
 - Each port on card is a unique CHPID (**1 PCHID: 1 Port relationship**)
 - *Multiple protocols can share the same physical port*
- Each port can be configured to provide a single function or combination of functions
- Support for OSE has been discontinued on z17

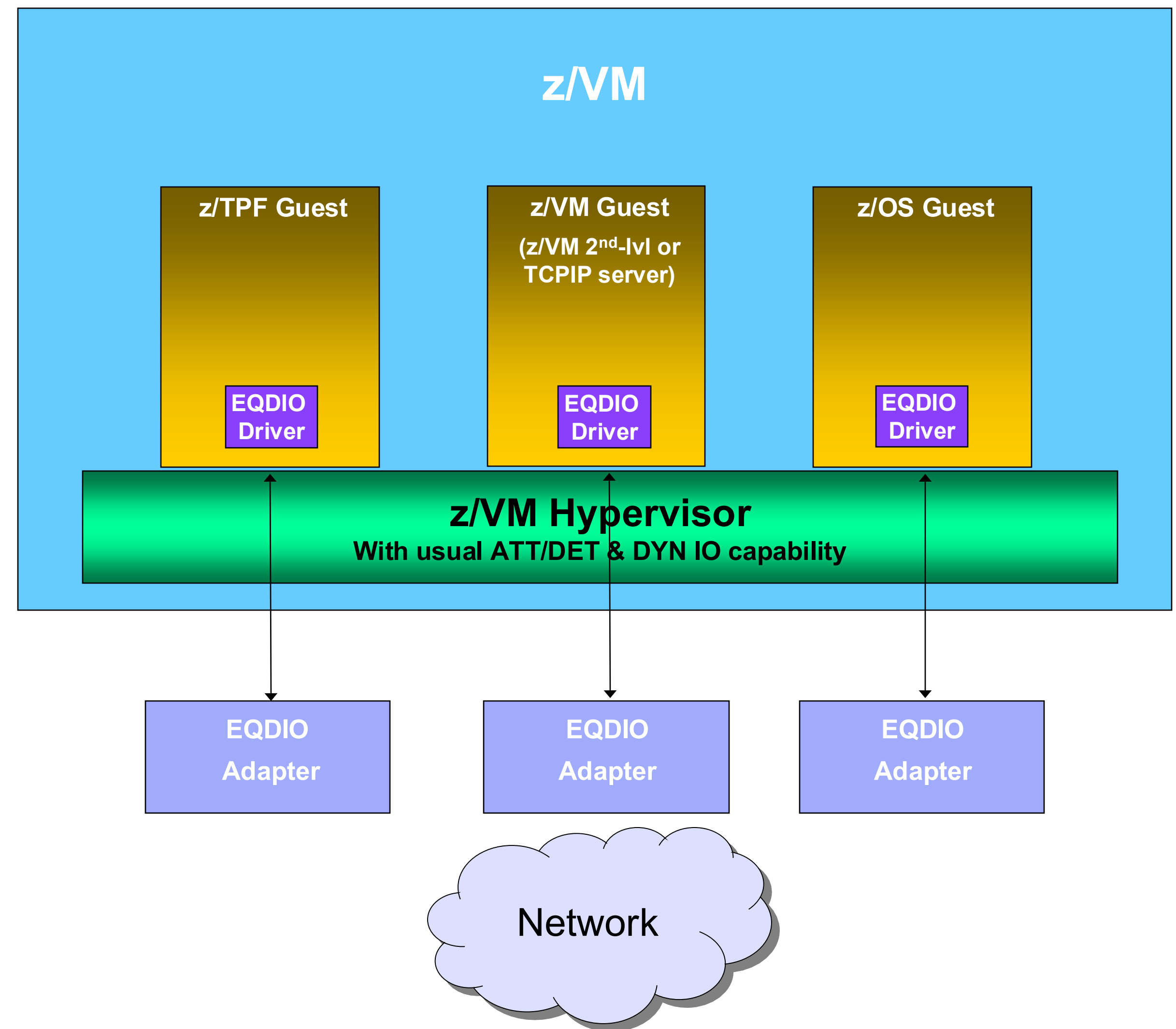


EQDIO and VSWITCH



Guest Support for Real EQDIO Devices

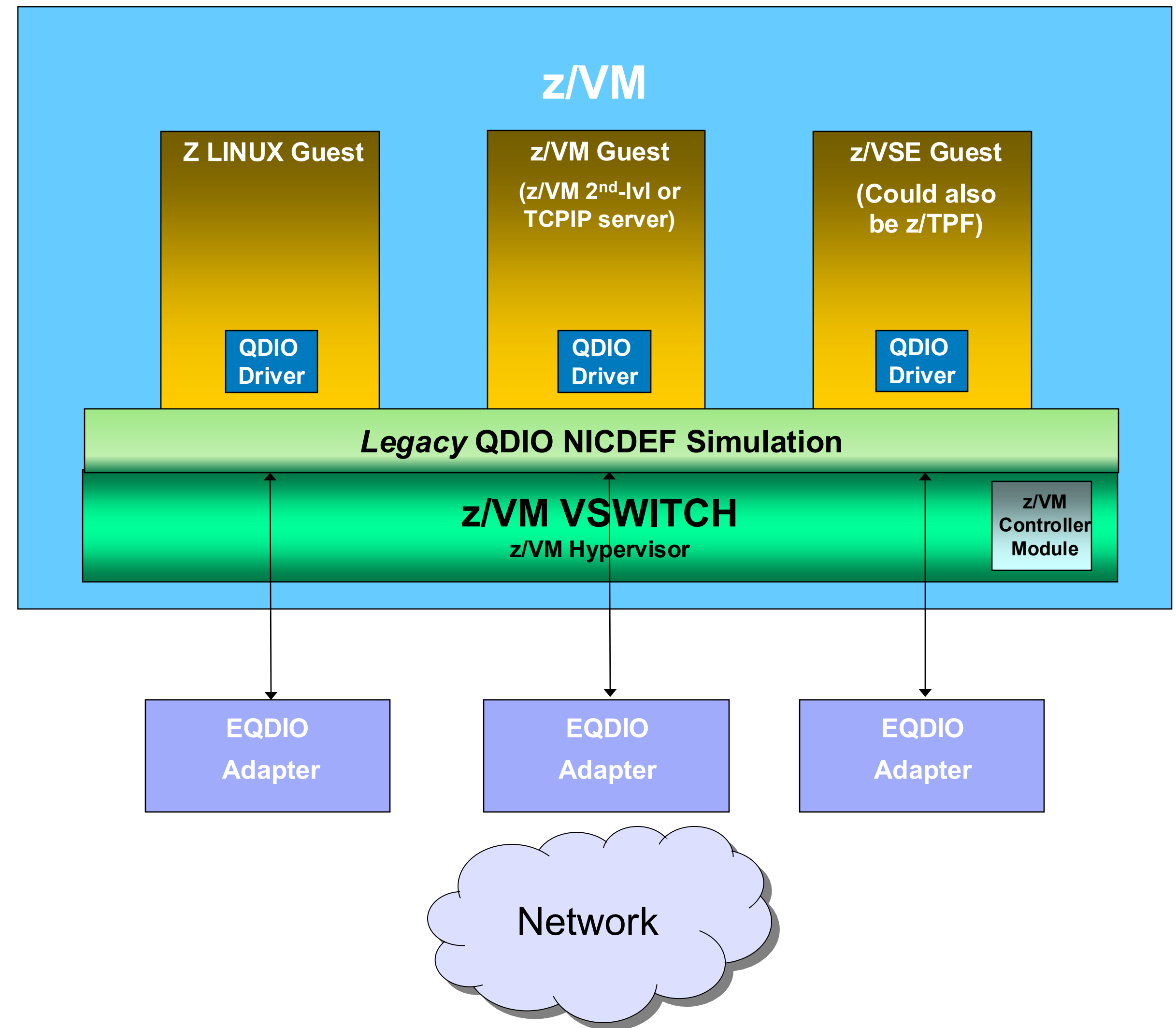
- Notice there is *no VSWITCH controller!*
- Usual Dynamic I/O support via HCD or CP **DEFINE** / **DELETE** / **MODIFY** commands
- Providing customary ability to **ATTACH** & **DETACH** EQDIO devices to guest virtual machines for their exploitation
- Similar **QUERY** virtual and real output as traditional QDIO
- We are providing *z/VM TCPIP virtual machine exploitation*
 - A firmware assist is also available for z/VM TCPIP



EQDIO VSWITCH Support

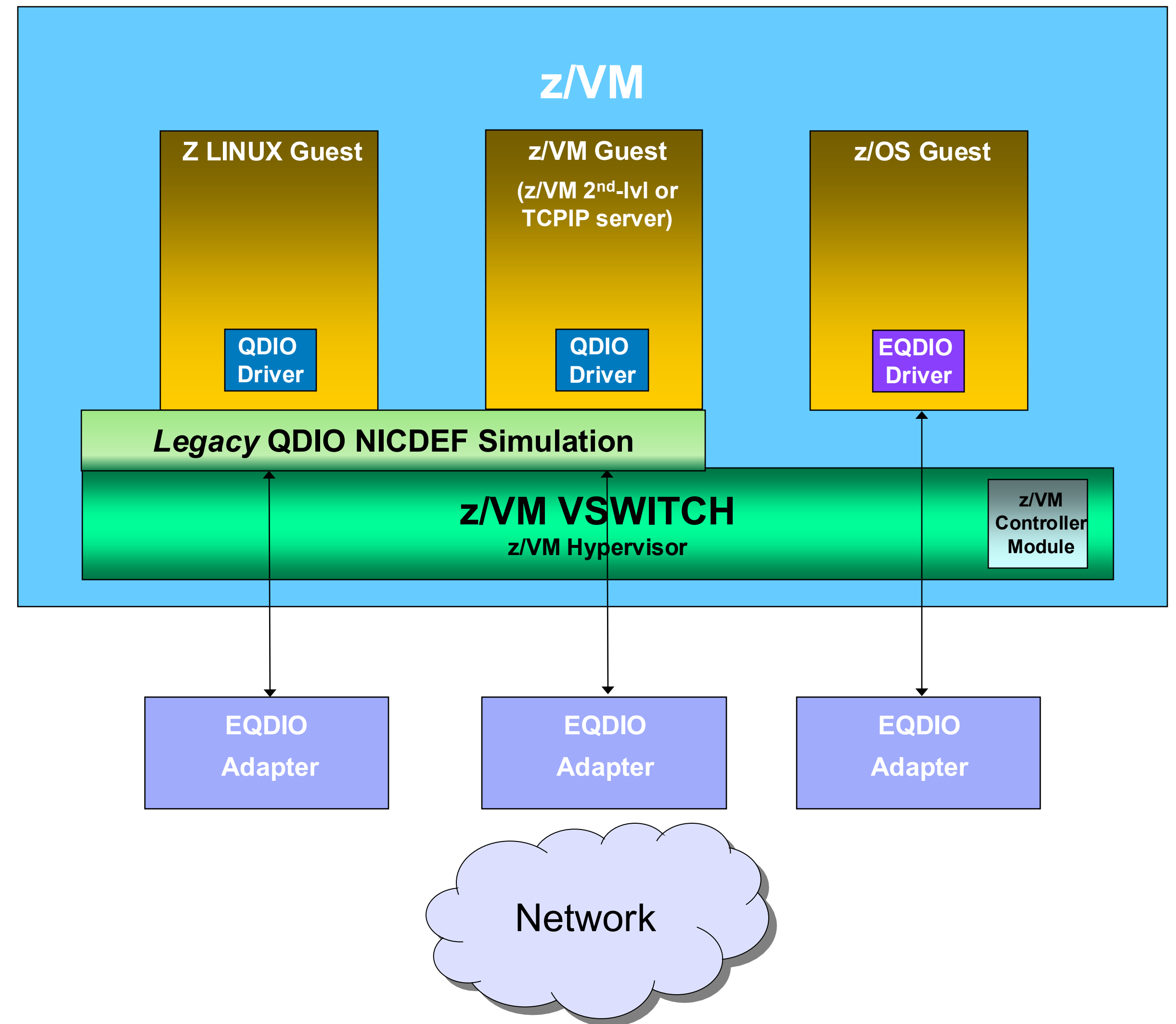
Guest OSes without native EQDIO support can take advantage of EQDIO hardware!

- VSWITCH Controller is provided by the VM Hypervisor rather than separate virtual machines
- z/VM VSWITCH supporting traditional QDIO-based simulated LAN to real EQDIO adapters (Uplink port)
- Guests use **DEFINE NIC TYPE QDIO**
- *EQDIO supports Layer 2 mode (ETHERNET) only*
- No Layer 3 (IP) support for EQDIO, unlike QDIO



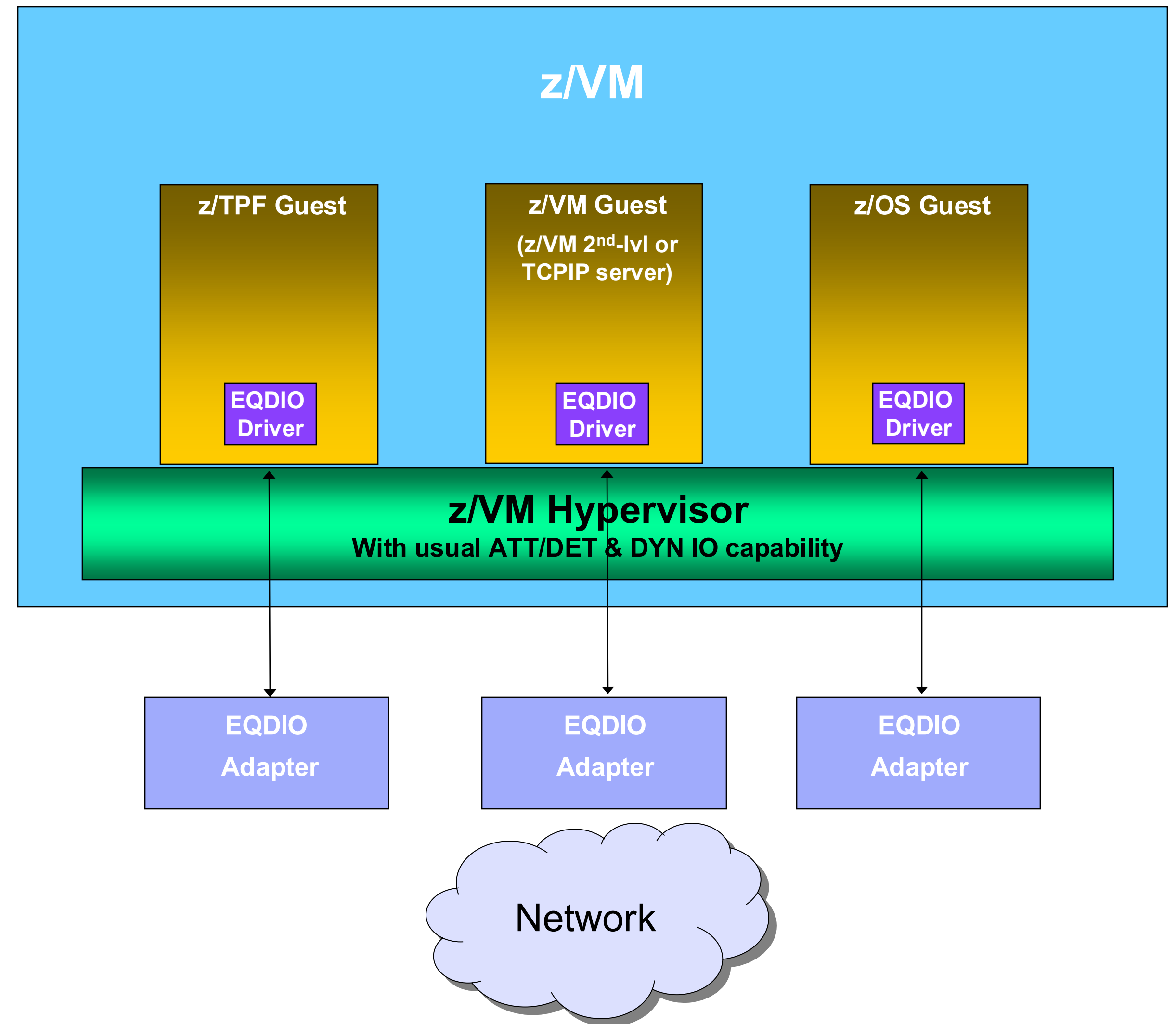
Which OS Needs which Device? EQDIO or QDIO?

- Linux cannot use an EQDIO VNIC; is not supported
- z/OS cannot support a QDIO VNIC connected to an EQDIO RDEV when configured as Layer 2; no plans to add support
- Since the VSwitch in diagram uses EQDIO uplink ports, z/OS is forced to use an EQDIO RDEV
- *z/OS requires an EQDIO RDEV, not VDEV*
- *Linux requires a QDIO VDEV, not EQDIO*
- Cross-communication support is facilitated by the VSwitch and simulation code



Guest Support for Real EQDIO Devices Migration Notes

- OSA device triplet has been reduced to a single **device** supporting both control and data functions. **Can leave triplets** and 2 DEVs will remain unused
- **No QIOASSIST or Live Guest Relocation (LGR)** function as there are *no plans for Linux to support EQDIO*

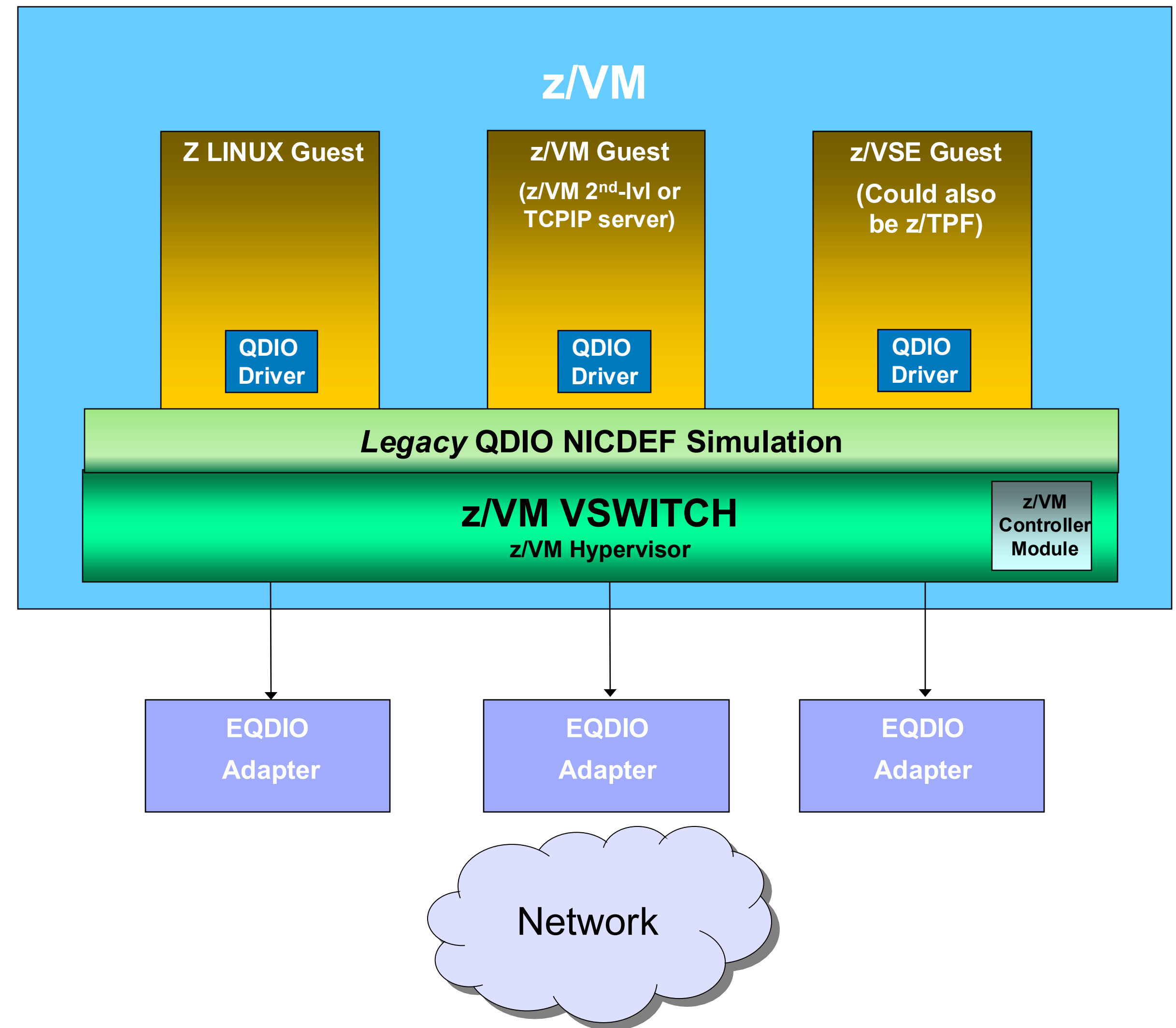


EQDIO VSWITCH Support External and Migration Notes

VSWITCH config externals remain mostly unchanged; some parameters not applicable to EQDIO

Migration Notes:

- Can leave original RDEV triplets and Hypervisor will figure out how to use them with EQDIO
 - This provides transparent migration for guests that support EQDIO
- Also provides LGR capability for Linux guests
- Protocol conversion from QDIO to EQDIO provided by VSWITCH will allow Linux to exploit EQDIO

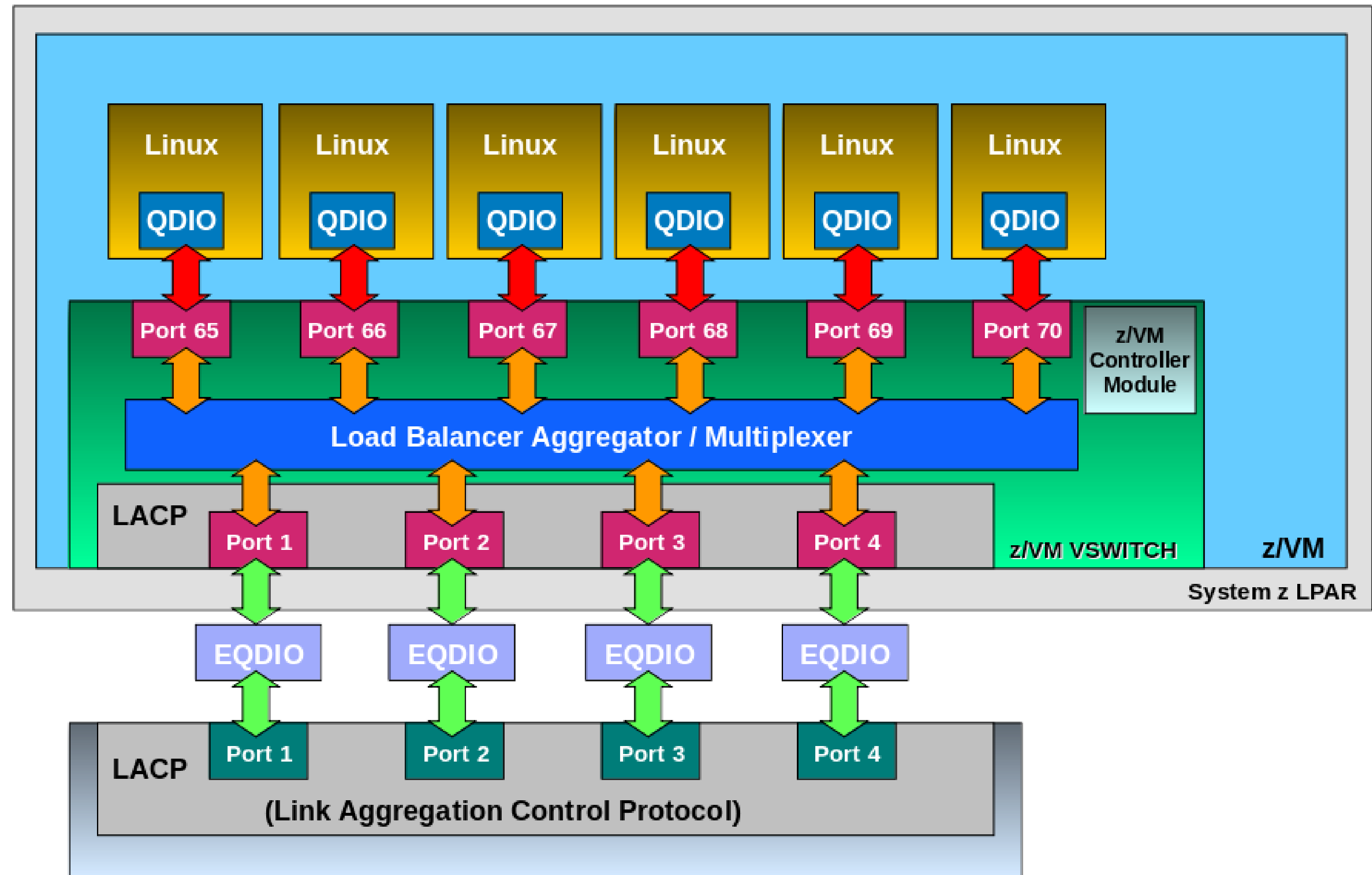


Link Aggregation using EQDIO

Non-disruptive networking
scalability & failover to redundant
ports

Still supporting up to **8 real EQDIO
ports** in aggregate (only 4 shown)

- *Cannot mix and match EQDIO
and QDIO in a port group*
- For OSH devices used in a Link
Aggregation port group, the
*IOCDS definition should include
CHPARM=02*
- This will only allow the device to
be configured in a LAG

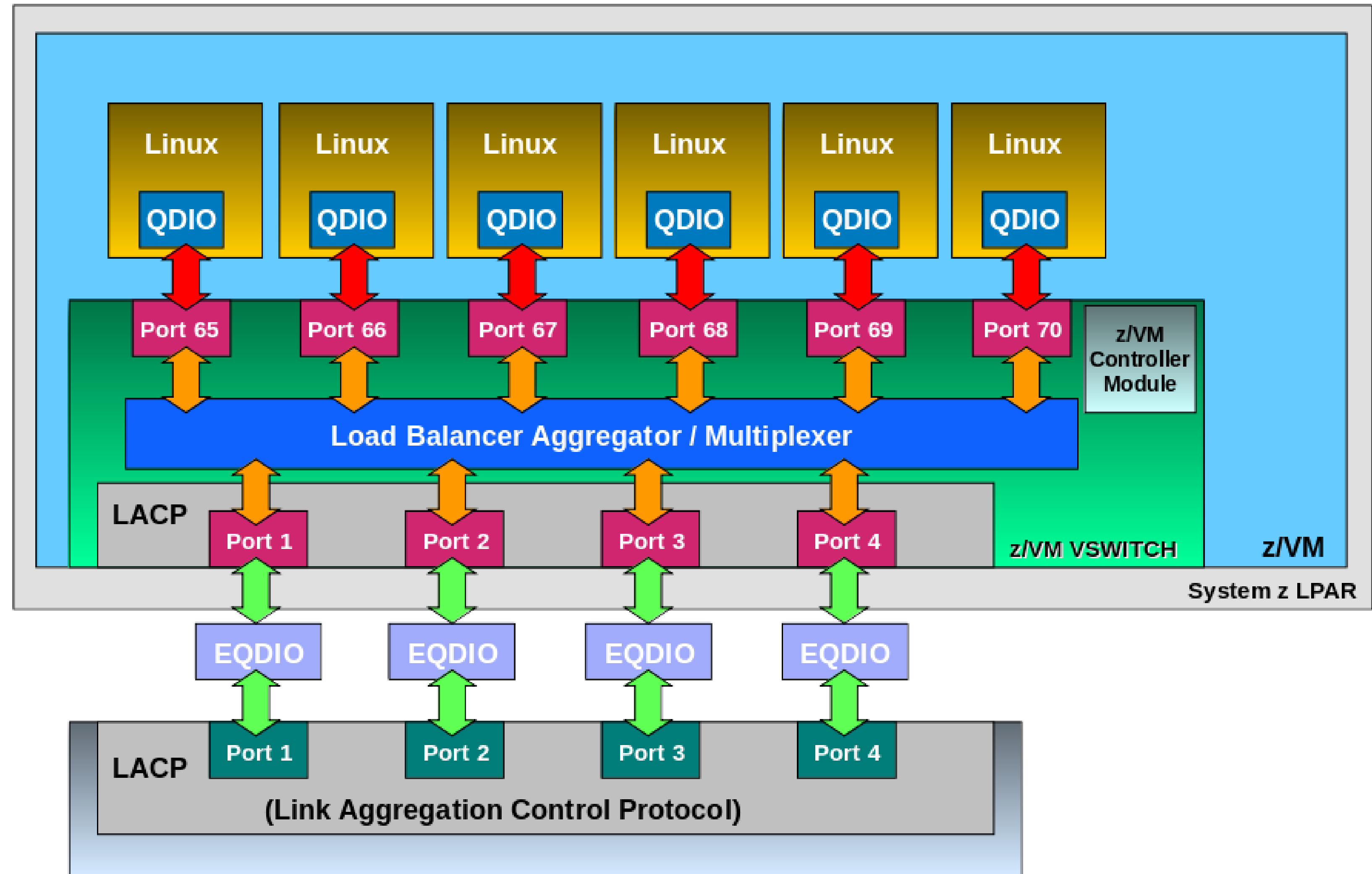


Link Aggregation using EQDIO - Continued

When a CHPID is **NOT** defined in **Link Aggregation mode**, attempts to bring up a VSWITCH that is configured to use the device as part of the **LAG Port Group** will **FAIL** if either:

1. A NETH device is also configured on the port, or
2. The system is DPM managed (Dynamic Partitioner Manager)

Confirm your system and device settings if problems are encountered during setup of new (non-migrated) LAGs

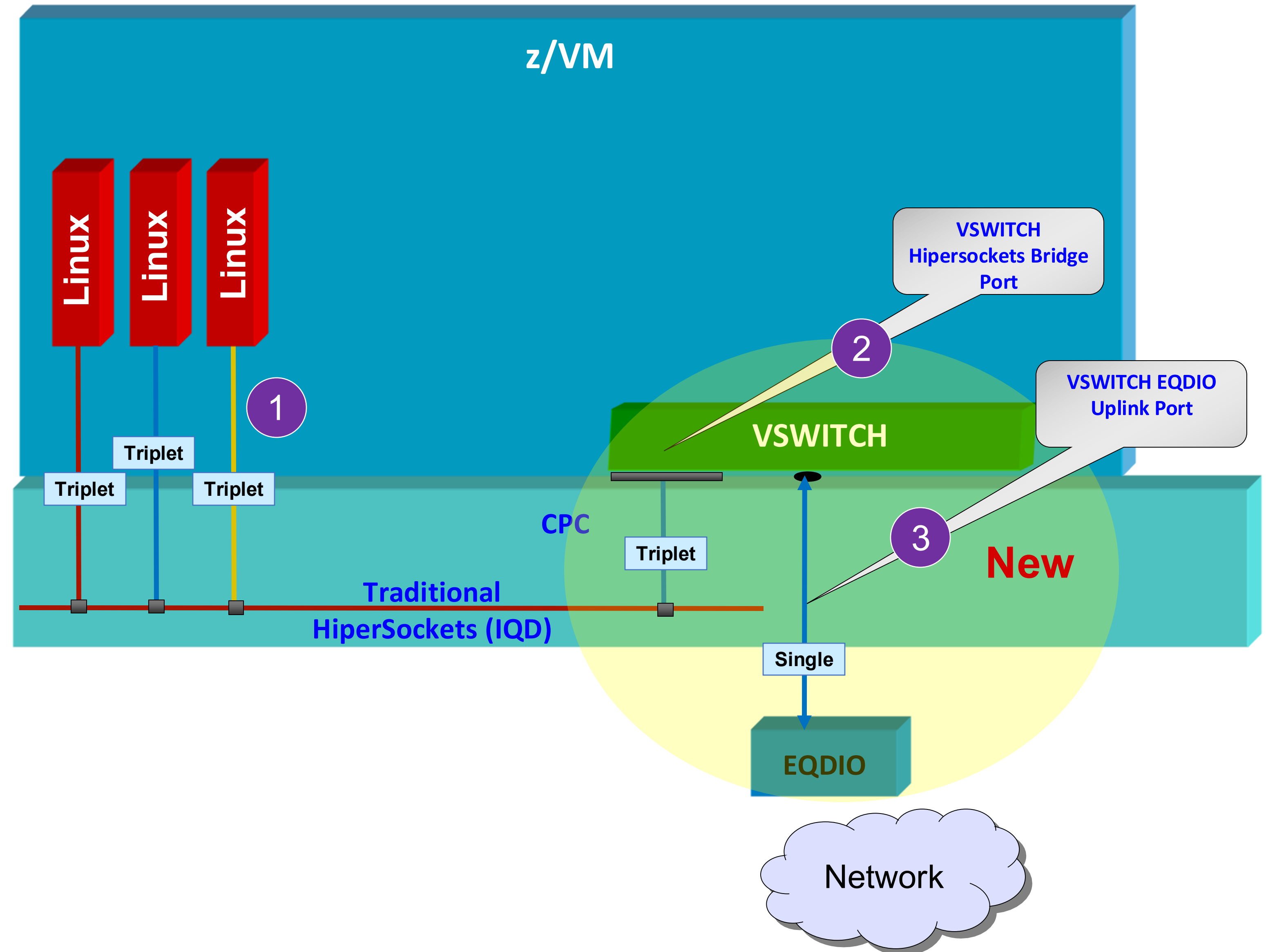


Hipersockets Bridge using EQDIO

Connection through z/VM VSWITCH with EQDIO adapter capability

- 1 ATTACH'd **real Hipersockets devices** exploiting QIOASSIST
- 2 Real Hipersockets bridged to **simulated LAN** (via Bridge port) with connectivity to external network (via uplink port)
- 3 Uplink port connection is in **VSWITCH with EQDIO adapter** (providing IQD to EQDIO conversion)

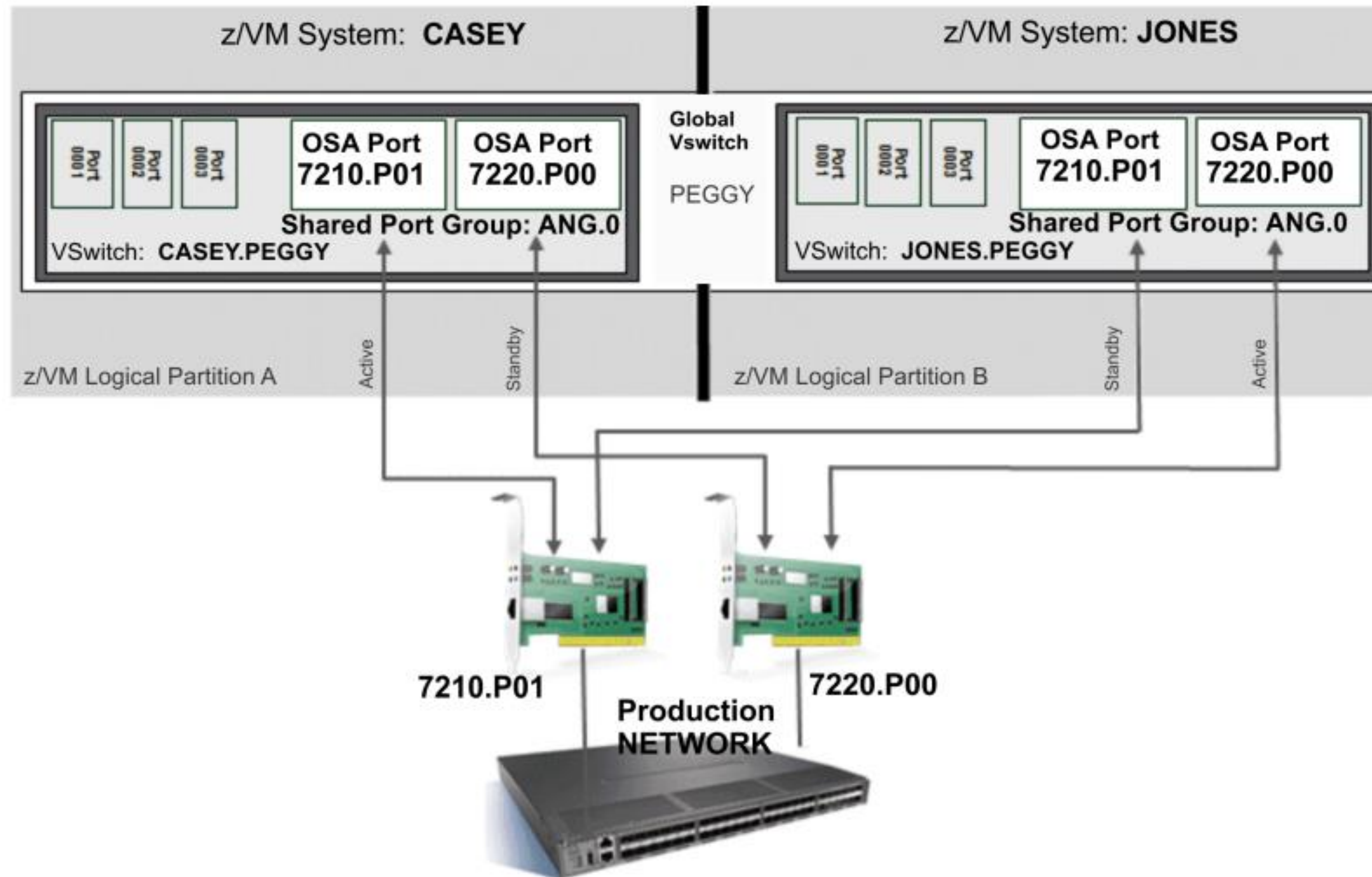
Link Aggregation (LAG) ports also supported (not shown)



Feature Pack Additions



Features Released through Feature Pack 3 Fix 00 – EQDIO MVLAG



APAR VM66877: Support for EQDIO MVLAG (Multi-VSwitch Link Aggregation Group)

- Functions essentially the same as QDIO MVLAG
- *Transparent migration* same as previously discussed; see Link Aggregation Using EQDIO slide
- *Use MVLAG like before*, but with the advantages of the EQDIO protocol
- *No changes to commands from QDIO MVLAG*

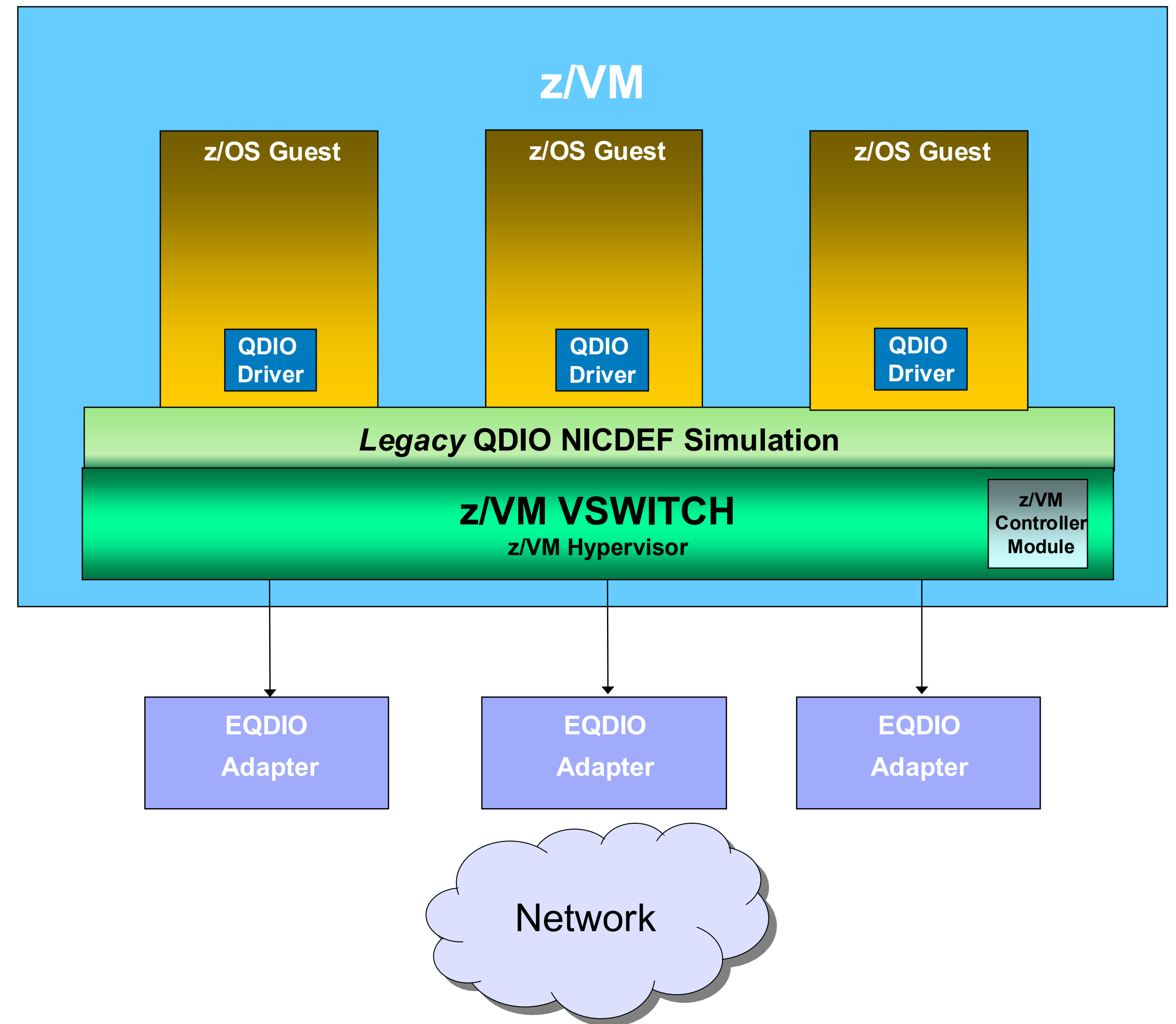
Features Released through Feature Pack 3 Fix 00 – Simulated EQDIO VNICs

APAR VM66877: Support for Simulated EQDIO VNICs

- Create a simulated EQDIO device for guest attachment to a VSwitch (*EQDIO VNICs can only be attached to QDIO Ethernet VSwitches*)
- Required for z/OS guests to connect to a VSwitch with an OSH uplink port (Network-Express)
- Externals for **DEFINE NIC** and **NICDEF** have been updated with **TYPE EQDIO**
- When a type EQDIO NIC is coupled to a QDIO VSwitch, CP utilizes the simulation code to provide the QDIO interface to a guest, while utilizing the EQDIO hardware

Features Released through Feature Pack 3 Fix 00 – Simulated EQDIO VNICs – Continued

- Simulated EQDIO adapters and simulated QDIO adapters can coexist on the same QDIO VSwitch
- **DEFINE NIC 2025 TYPE EQDIO**
- **NICDEF 2025 TYPE EQDIO**
- **COUPLE 2025 TO SYSTEM NETWORK** (Given that NETWORK is a QDIO VSwitch owned by SYSTEM)



Features Released through Feature Pack 3 Fix 00 – Simulated EQDIO VNICs Continued

Possible Issue:

- Guest z/VM systems can experience device initialization errors on certain configurations using simulated devices. These devices may receive the message:

"HCPPIO6284I Device nnnn cannot be varied online due to conflicting device classification information."

- *This has been fixed in the z/VM 7.4 CP Fix Pack 0301.*

Features Released through Feature Pack 4 Fix 01 – **Please Install!!!**

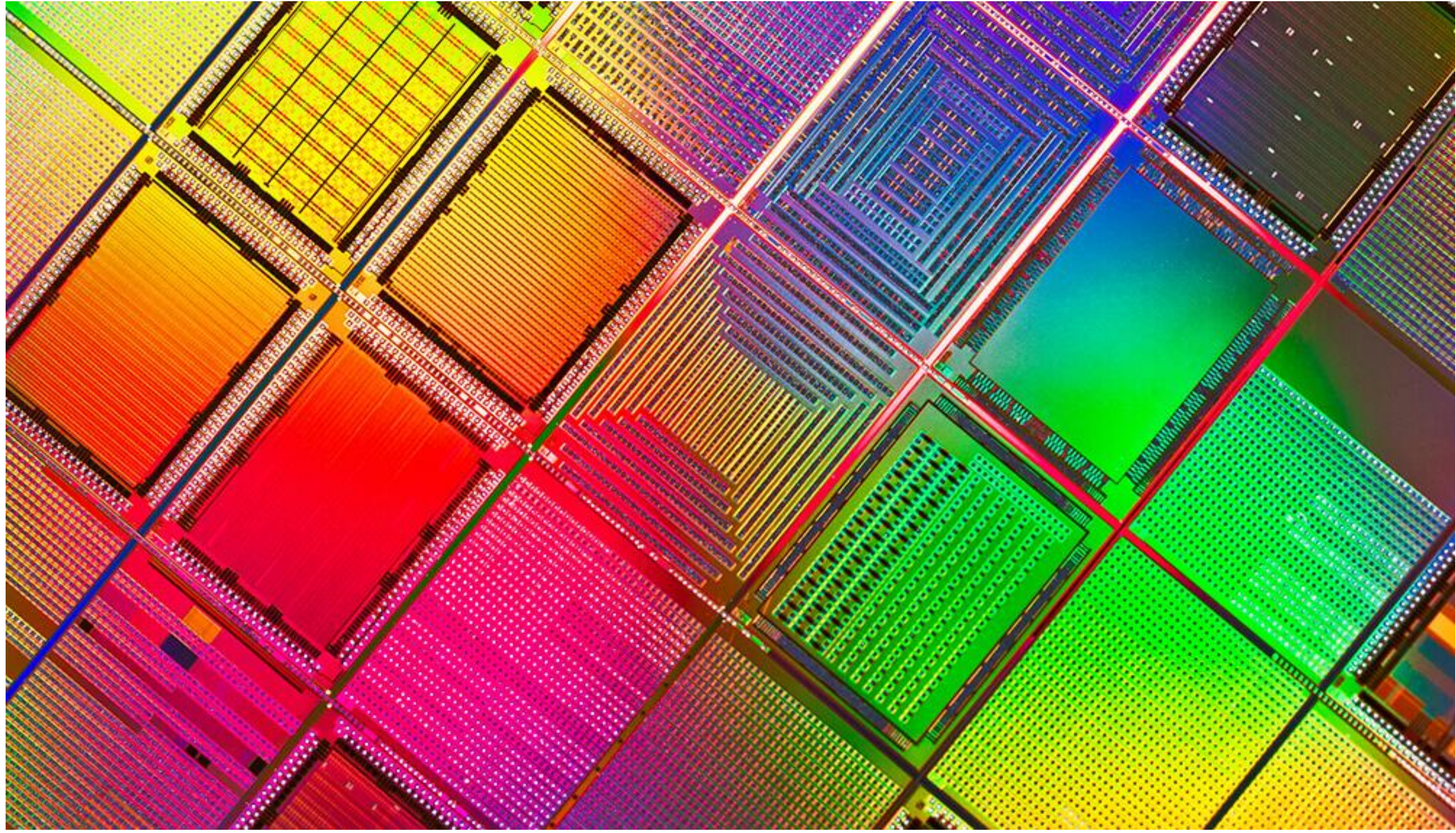
PLEASE INSTALL!

APAR VM66939: EQDIO and VSwitch Critical Fixes (Newest Firmware Bundle REQUIRED!)

- Reduces EQDIO queue interrupts that can cause delays in packet processing
- "Nearest Neighbor Switch" capabilities added to QUERY OSAINFO command output
- Enforces ISOLATION can only be turned ON for an EQDIO VSwitch when the VSwitch is DISCONNECTed, with a CONNECT required to bring up the VSwitch in ISOLATE mode
- QUERY OSAINFO RDEV command for a BACKUP device error message output clarification
- Enable VSwitch to dynamically change RDEVs when all RDEV switch ports are down
- IVG001 ABEND following Global VSwitch detach resolved

PLEASE INSTALL!

Externals



Changes to MONITOR Records

One *new entire record for EQDIO data*, similarly structured to the QDIO one in order to provide redundancy and diagnostics

MRIODVSW, *original Virtual Switch Activity Record*, now has parameters to track storage range and current values for **QUEUESTORAGE** parameter

MRIODVSE, *new Virtual Switch EQDIO Activity Record* to collect *data useful for EQDIO devices and queue activity*

- Control plane byte and packet counters
- Data plane event counters
- EQDIO RX, TX queue byte counts, etc.

Note: *Certain VSwitch data may still need to be pulled from MRIODVSW*. EQDIO queue-specific information is located in MRIODVSE.

Query OSAINFO Command

This command was added as *additional diagnostic output*
Previously provided by the VSWITCH Controller

```
QUERY OSAINFO VSW1 DETAILS
VSWITCH SYSTEM  VSW1    RDEV 6049
OSA Generation:      Network Express
OSA Firmware Level:   x3031240088800008
Card Type:           10G Ethernet
Active Speed:         10G
Link State:           Online
Port Media Type:      Multi-Mode Fiber
Adapter VCHID number: x0410
MTU Size:             9000
Max MTU Size:         9000
PCHID:                x01B8
CHPID:                x0068
TX Control Plane MTU: Max data transfer 4K
RX Control Plane MTU: Max data transfer 64K
Ethernet Burned-in-Address: 9C-63-C0-0B-64-22
Current Ethernet MAC Address: 9C-63-C0-0B-64-22
NIC Capabilities:     All Promiscuous Mode
NIC Capabilities:     Unicast Prom Mode
NIC Capabilities:     Multicast Prom Mode
NIC Capabilities:     LPAR to LPAR
NIC Capabilities:     Drop Light
NIC Capabilities:     LAG
NIC Capabilities:     Reflective Relay
NIC Capabilities:     Jumbo Frames
Maximum number of TX Queues: 4
Highest TX Queue Priority: 1
Lowest TX Queue Priority: 4
```

```
8 Byte SBPE Allocation Multiple: 512
16 Byte SBPE Allocation Multiple: 256
RX Buffer Allocate Size: 32
Max Service Units TX P1: 2048
Max Service Units TX P2: 1024
Max Service Units TX P3: 512
Max Service Units TX P4: 256
RX Latency Timer Resolution: 250
RX Streaming Timer Resolution: 2000
Port MAC address: 74-99-75-C5-21-00
MAX Frame Size: 0
System Description: System Description = IBM Networking
System Description: Operating System RackSwitch G8264, IBM
System Description: Networking OS: version 7.11.12, Boot
System Description: image: version 7.11.12.
Port Description: Port Description = 41.
Switch Name: END8264-6.
Interface Name: , Id: (hex) 3431.
Device Number: x6049
Control Unit Logical Address: x00
Unit Address: x02 Number Of Output Queues: 5
Number Of Input Queues: 2
Number Of Active Input Queues: 1
Default RX Queue ID: x0002
Interface Activation Attr Flags: Shared
Nbr of Active Hosts: 0
Interface Drop Count: 0
-----
VMAC Address: 02-00-00-00-00-02 Host Generated
-----
Multicast MAC Address: 01-00-5E-00-00-01
```



SET VSWITCH LOG Command

Previously with QDIO, diagnostic output for uplink devices could be viewed on the VSWITCH controller log

With the embedded controller for EQDIO, this has been implemented differently

```
SET VSWITCH <VSWITCHNAME> LOG ON
SET VSWITCH <VSWITCHNAME> LOG OFF
SET VSWITCH <VSWITCHNAME> LOG VERBOSE
```

OFF

```
SET VSWITCH TEST CONNECT
17:52:43 Command complete
17:52:43 HCPSWU2830I VSWITCH SYSTEM TEST status is ready.
17:52:43 HCPSWU2830I SYSTEM is VSWITCH controller for device 6746.
Ready OPERATOR at 2NDMLEE2 ;
```

ON

```
SET VSWITCH TEST CONNECT
17:53:47 Command complete
Ready OPERATOR at 2NDMLEE2 ;
17:53:47 HCPSWU2830I VSWITCH SYSTEM TEST status is ready.
17:53:47 HCPSWU2830I SYSTEM is VSWITCH controller for device 6746.
17:53:47 HCPSWU3227I Device: 6746 Command: Initial Data Exchange RC: 0000
17:53:47 HCPSWU3227I Successful Completion
17:53:47 HCPSWU3227I Device: 6746 Command: Start a LAN Adapter RC: 0000
17:53:47 HCPSWU3227I Successful Completion
17:53:47 HCPSWU3227I Device: 6746 Command: Set Interface Parms RC: 0000
17:53:47 HCPSWU3227I Successful Completion
17:53:48 HCPSWU3227I Device: 6746 Command: Activate_Data_Queues RC: 0000
17:53:48 HCPSWU3227I Successful Completion
```

SET VSWITCH LOG Command – VERBOSE Output

```
SET VSWITCH TEST CONNECT
17:49:10 Command complete
Ready OPERATOR at 2NDMLEE2 ;
17:49:10 HCPSWU2830I VSWITCH SYSTEM TEST status is ready.
17:49:10 HCPSWU2830I  SYSTEM is VSWITCH controller for device 6746.
17:49:10 HCPSWU3227I Device: 6746  Command: Initial Data Exchange RC: 0000
17:49:10 HCPSWU3227I Successful Completion
17:49:10 HCPSWU3227I +0000 10000001 00000100 13000000 00000000
17:49:10 HCPSWU3227I +0010 00000000 0002B101 00000000 00000000
17:49:10 HCPSWU3227I +0020 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I +0090 00000000 D585A3A6 96999240 C5A79799
17:49:10 HCPSWU3227I +00A0 85A2A200 00000000 00000000 00000000
17:49:10 HCPSWU3227I +00B0 00000000 31319900 00000000 00100008
17:49:10 HCPSWU3227I +00C0 00010002 04062328 00002328 01340079
17:49:10 HCPSWU3227I +00D0 00040040 B1010000 00000000 00000000
17:49:10 HCPSWU3227I +00E0 946DAEF6 61800000 946DAEF6 61800000
17:49:10 HCPSWU3227I +00F0 00000000 0000003F 00000000 00000000
17:49:10 HCPSWU3227I +0100 04010400 02000100 00000020 08000400
17:49:10 HCPSWU3227I +0110 02000100 00000000 00FA07D0 00000000
17:49:10 HCPSWU3227I +0120 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I +0290 00000000 749975C5 21000000 00000000
17:49:10 HCPSWU3227I +02A0 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +02B0 00000000 E2A8A2A3 859440C4 85A28399
17:49:10 HCPSWU3227I +02C0 8997A389 9695407E 40C9C2D4 40D585A3
17:49:10 HCPSWU3227I +02D0 A6969992 89958740 D6978599 81A38995
17:49:10 HCPSWU3227I +02E0 8740E2A8 A2A38594 40D98183 92E2A689
17:49:10 HCPSWU3227I +02F0 A3838840 C7F8F2F6 F46B40C9 C2D440D5
17:49:10 HCPSWU3227I +0300 85A3A696 99928995 8740D6E2 7A40A585
17:49:10 HCPSWU3227I +0310 99A28996 9540F74B F1F14BF1 F26B40C2
17:49:10 HCPSWU3227I +0320 9696A340 89948187 857A40A5
8599A289
17:49:10 HCPSWU3227I +0330 969540F7 4BF1F14B F1F21500 00000000
17:49:10 HCPSWU3227I +0340 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I +03B0 00000000 D79699A3 40C485A2 83998997
17:49:10 HCPSWU3227I +03C0 A3899695 407E40F2 F5150000 00000000
17:49:10 HCPSWU3227I +03D0 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I +0410 00000000 C5D5C4F8 F2F6F460 F6150000
17:49:10 HCPSWU3227I +0420 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I +0450 00000000 6B40C984 7A404D88 85A75D40
17:49:10 HCPSWU3227I +0460 F3F2F3F5 15000000 00000000 00000000
17:49:10 HCPSWU3227I +0470 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I +....
17:49:10 HCPSWU3227I Device: 6746  Command: Start a LAN Adapter RC: 0000
17:49:10 HCPSWU3227I Successful Completion
17:49:10 HCPSWU3227I +0000 01000002 00000100 13000000
17:49:10 HCPSWU3227I Device: 6746  Command: Set Interface Params RC: 0000
17:49:10 HCPSWU3227I Successful Completion
17:49:10 HCPSWU3227I +0000 B7000003 00000100 13000000 00000000
17:49:10 HCPSWU3227I +0010 00000000 001C0000 00000002 00000000
17:49:10 HCPSWU3227I +0020 00000000 00000000 00000000 00000000
17:49:10 HCPSWU3227I Device: 6746  Command: Activate_Data_Queues RC: 0000
17:49:10 HCPSWU3227I Successful Completion
17:49:10 HCPSWU3227I +0000 11000005 00000100 13000000
```

SET/DEFINE VSWITCH ISOLATION Command (EQDIO)

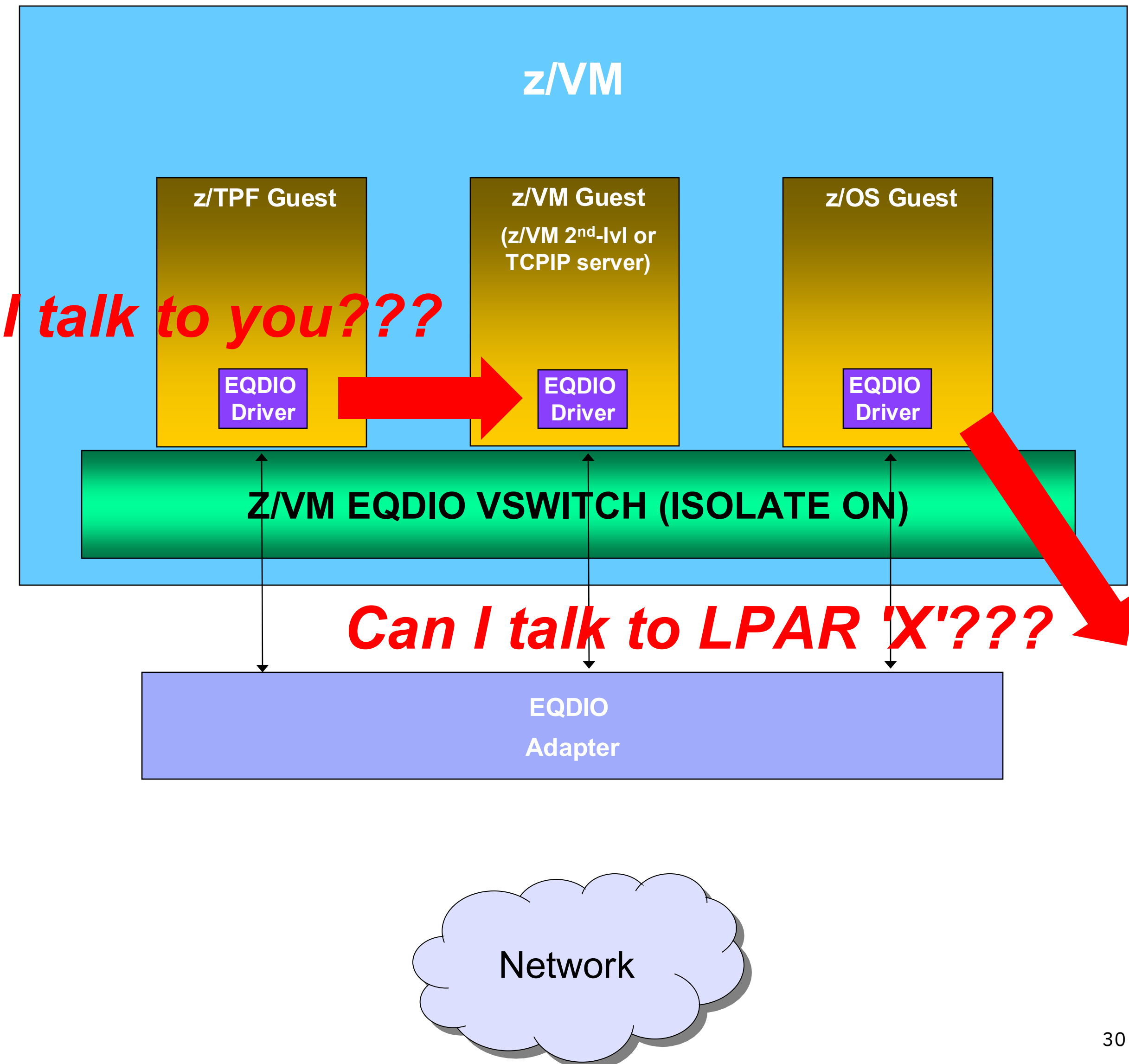
- **New parameter on DEFINE VSWITCH, only for EQDIO VSwitches!** This was released in pack Feature 04 Fix 01.
- When **ON**, *guests are prohibited from sending traffic to other guests on the same VSwitch*. No direct LPAR communications sharing the same OSA port is permitted for the guest ports. Any traffic destined for the VSwitch guest ports from LPARs sharing the same OSA port will be dropped.
- Default ISOLATION mode is OFF
- Not supported on a VSwitch with Hipersockets Bridge Port
- ISOLATION ON can also be done with **SET**, *but cannot be done dynamically, like it was with QDIO*
- *If the ISOLATION mode is changed with SET, it must be done while DISCONNECTed*, otherwise the ISOLATION state will not change until the VSwitch is DISCONNECTed and re-CONNECTed.
- *EQDIO hardware does not support dynamic ISOLATION mode changes*, unlike QDIO

SET/DEFINE VSWITCH ISOLATION Command (EQDIO) - Continued

- DEFINE VSWITCH TESTLAG1 ETHERNET ISOLATION ON UPLINK RDEV 5555
- Given that 5555 is an EQDIO device, VSwitch TESTLAG1 will come up (CONNECT) after DEFINE as an isolated EQDIO VSwitch

OR!

- DEFINE VSWITCH TESTLAG2 ETHERNET UPLINK RDEV 5555
- SET VSWITCH TESTLAG2 DISCONNECT
- SET VSWITCH TESTLAG2 ISOLATION ON
- SET VSWITCH TESTLAG2 CONNECT
- TESTLAG2 will CONNECT and establish an ISOLATED connection

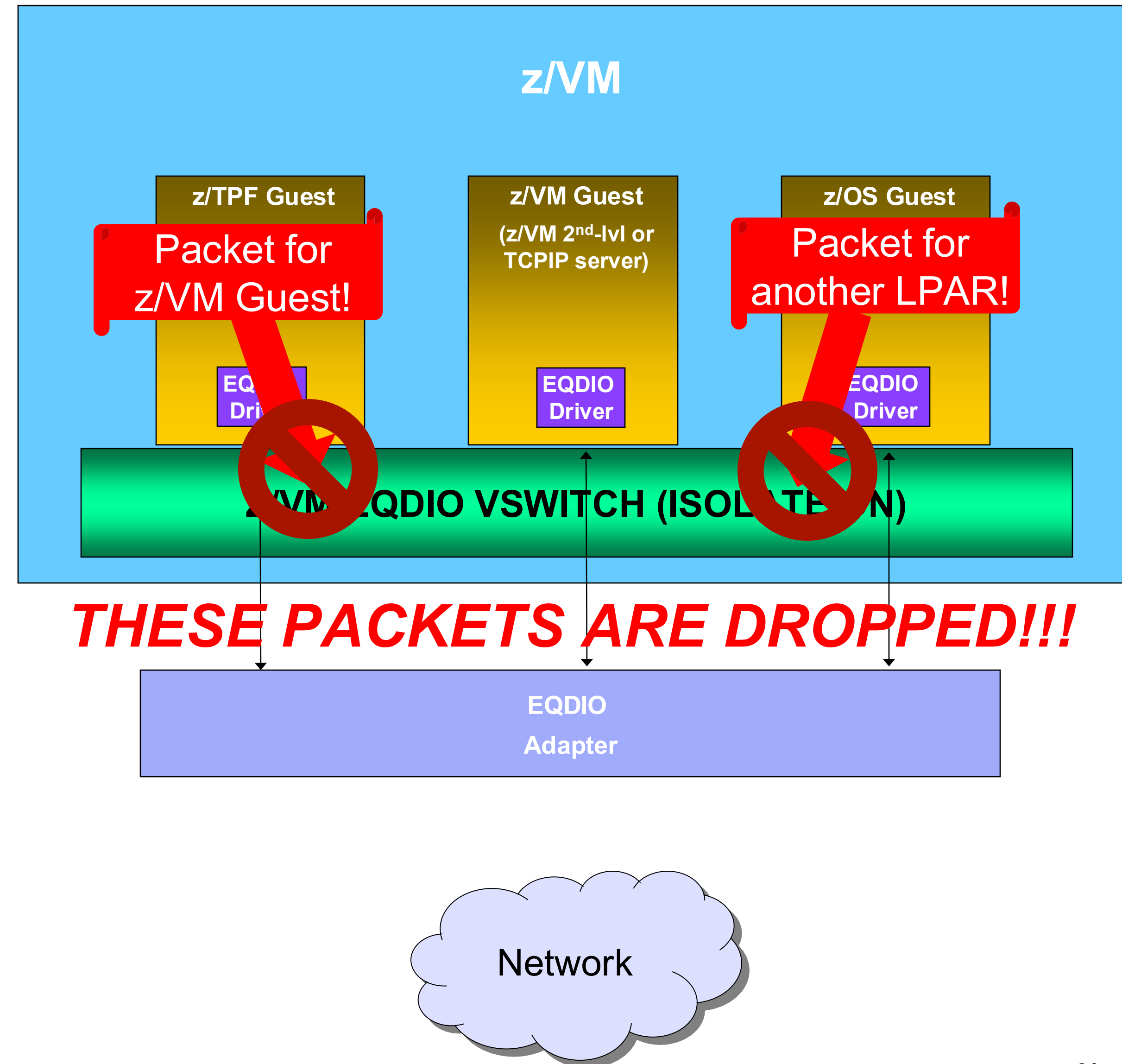


SET/DEFINE VSWITCH ISOLATION Command (EQDIO) - Continued

- DEFINE VSWITCH TESTLAG1 ETHERNET ISOLATION ON UPLINK RDEV 5555
- Given that 5555 is an EQDIO device, VSwitch TESTLAG1 will come up (CONNECT) after DEFINE as an isolated EQDIO VSwitch

OR!

- DEFINE VSWITCH TESTLAG2 ETHERNET UPLINK RDEV 5555
- SET VSWITCH TESTLAG2 DISCONNECT
- SET VSWITCH TESTLAG2 ISOLATION ON
- SET VSWITCH TESTLAG2 CONNECT
- TESTLAG2 will CONNECT and establish an ISOLATED connection



TCP/IP on z/VM



z/VM TCP/IP and EQDIO Configuration

A new **DEVICE** statement added for EQDIO configuration in the **PROFILE TCPIP**

- This is where EQDIO specific device configuration options are located

IPWIZARD has been updated to **support configuration of EQDIO adapters** during z/VM installation

- New install panel for EQDIO

IFCONFIG has been updated to **support dynamic configuration of EQDIO adapters** *after TCP/IP has been initialized*

z/VM TCP/IP and EQDIO Configuration – Continued

New parameters on **EQDIO DEVICE** statement:

- **BUFFERS** – defines the number of input buffers available to the adapter

Parameters from OSD DEVICE statement *not* used for EQDIO:

- **PORTNAME**
- **PORTNUMBER**
- **PRIROUTER / SECROUTER / NONROUTER**
- **IPV6PRIROUTER / IPV6SECROUTER / IPV6NONROUTER**
- **IGNORESETROUTERERRORS**

z/VM TCP/IP and EQDIO Configuration – Continued

New **EQDIO DEVICE** statement:

```
DEVICE eth1 EQDIO 600 BUFFERS 2048
```

LINK statement will use type **EQDIOETHERNET** for EQDIO

Options the same as **QDIOETHERNET** except: IP/ETHERNET option no longer used
(**ETHERNET is the only valid value**)

DEDICATE or ATTACH statement only need specify a single device address for EQDIO adapters

z/VM TCP/IP and EQDIO Migration

Allow for use of previous OSD device statements for EQDIO adapters:

```
DEVICE eth1 OSD 600 PORTNAME qdioport PORTNUMBER 2  
NONROUTER AUTORESTART
```

If device 600 is redefined as an EQDIO adapter, TCP/IP will initialize it as such, **ignoring parameters that are obsolete for EQDIO**

z/VM TCP/IP and EQDIO Migration

Note that **previous DEDICATE or ATTACH statements** would use **three OSA addresses** (read, write, data) for QDIO. While the same statements will continue to work, *only one device address is needed for EQDIO*.

Allows the use of previous z/VM TCP/IP configuration files without modification

Summary

z17 brings modernization to networking infrastructure on IBM Z by moving I/O functionality from adapter cards to the Z processor, **enhancing performance, latency, bandwidth, and scalability** while maintaining backward compatibility

The EQDIO protocol supports **multiple networking protocols on converged adapters**, reducing hardware complexity and **improving resource utilization**.

*Since its release, we have been **adding features and making improvements to the reliability and servicability of our EQDIO support**, with a focus on LAG.*

Thank you!

Montana Lee

z/VM Networking Developer

montanalee@ibm.com

<https://www.linkedin.com/in/montanaalexandralee/>

Special Thanks To...

- Miguel Delapaz
- Dennis Musselwhite
- Walter Church IV
- Richard Tarzca
- Steve Wilkins
- Frederik Hartmann
- Caroline Hart

References

z/VM 7.4 CP Change Log – Service/Feature Packs

<https://www.vm.ibm.com/changelog/740/cp.html>

z/VM 7.4 Connectivity:

https://www.ibm.com/docs/en/SSB27U_7.4.0/pdf/hcpa6_v7r4.pdf

z/VM 7.4 TCP/IP Planning and Customization:

https://www.ibm.com/docs/en/SSB27U_7.4.0/pdf/kill0_v7r4.pdf

z/VM 7.4 VSWITCH EQDIO Activity MONITOR Record:

<https://www.vm.ibm.com/pubs/mon740/mriodvse.html>

z/VM 7.4 VSWITCH Activity (QDIO) MONITOR Record:

<https://www.vm.ibm.com/pubs/mon740/mriodvsw.html>

IBM Z Connectivity Handbook (Redbooks):

<https://www.redbooks.ibm.com/redbooks/pdfs/sg245444.pdf>